

STORCH AMINI & MUNVES PC
Russell D. Munves
Two Grand Central Tower, 25th Floor
140 East 45th Street
New York, NY 10017

BERGER & MONTAGUE, P.C.
Todd S. Collins
1622 Locust Street
Philadelphia, PA 19103
(215) 875-3000

Counsel for Plaintiff

**CONTAINS HIGHLY CONFIDENTIAL
INFORMATION SUBJECT TO
PROTECTIVE ORDER**

UNITED STATES DISTRICT COURT
SOUTHERN DISTRICT OF NEW YORK

-----X
ADVANCED ANALYTICS, INC., :

Plaintiff and Counterclaim Defendant, : No. 04 Civ. 3531 (LTS)(HBP)

v. :

CITIGROUP GLOBAL MARKETS, INC., :
SALOMON SMITH BARNEY, INC., and
THE YIELD BOOK INC., f/k/a SALOMON :
ANALYTICS, INC., :

Defendants and Counterclaim-Plaintiffs.

-----X

FOURTH DECLARATION OF JIANQING FAN

Contents

FOURTH DECLARATION OF JIANQING FAN	1
Summary of Findings	5
Top Level Description of ACE Theft Code and ACE Use Code Flow Chart	8
A. Defendants tampered with the ACE test result productions	13
B. None of the sequences produced by Defendants can be generated by The Yield Book	15
C. The non-existent Russell 1000 Path Sequence in reality was a code name in Defendants' code referring to a Super ACE sequence	24
D. Defendants' Deficient and Altered Sequence File and Testing File Production	33
D1. Defendants destroyed all the original code relating to seed selection algorithm	34
D2. Defendants destroyed all code files relating to sequence selection, and also variance reduction and testing for LDS100, many of which, as shown by Teytel's Notebook, have "ACE" in their names	35
D3. LDS200 files fabrication I: Teytel modified the metadata "Date Modified" of all LDS200 files	37
D4. Development Production fabrication II: Teytel destroyed all of the original sequence development files and testing files to cover up the fabrication	38
D5. LDS200 fabrication III: Teytel packed the Production with tens of thousands of redundant RCS files	39
D6. The Development Production was clearly fabricated using test files irrelevant to support the phony Teytel Algorithm	40
D7. Defendants destroyed all of Radak 1000 Path Sequence development files and testing files that would confirm that Russell 1000 Path Sequence does not exist. Defendants also withheld all ACE systems that would reveal their theft of ACE	43
E. The ACE Theft Code in The Yield Book	44
E1. The Theft Code that Defendants used to steal ACE	47
E2. Defendants prepared ACE Theft Code to steal ACE sequences, and then the next day invited Dr. Wang to do tests	51
E3. Defendants ACE Theft Code clearly specified that its goal was "save_seq" and that the sequences they intended to "save and reuse" was not their one and only sequence, the Production 200 in use when they were testing ACE, it was ACE	51
E4. There was no reason for Defendants to "save and reuse" their own sequence, only saving ACE sequences makes sense	53
E5. Defendants have confirmed that other than the ACE sequences and their own sequences, they never had access to any other sequences	54

E6. ACE Theft Code was customized to capture the entire length of an ACE sequence.	54
E7. The Theft Code was developed in lock steps with the progression of ACE tests	56
E8. Defendants used ACE Theft Code in the last ACE test and stole all ACE sequences.....	58
E9. Defendants' admission that they and others would be motivated to steal ACE and use it, reflects exactly what they did to obtain the means to pursue an MBS Arbitrage Strategy	59
F. Defendants surreptitiously switch to ACE <i>right after</i> they stole ACE in the last ACE test in June 1998 and relied on ACE for their production until at least 2007	61
F1 Right after the ACE theft, Defendants started installing ACE Use Code in The Yield Book, this code surreptitiously replaced "Defendants' sequence" by ACE sequences in production.....	62
F2. Right after the ACE theft, Defendants started installing new ACE Use Code in The Yield Book exclusively for Yield Book internal users including their own traders.....	69
F3. One month after ACE theft, Defendants installed the ACE Use Code for their trading and production in global financial markets.....	74
F4. Right after ACE theft, Defendants started installing new ACE Use Code to use ACE term structures for The Yield Book in production.....	76
F5. Right after the ACE theft, Defendants started compiling executable code using the ACE Use Code to build The Yield Book systems for The Yield Book production.....	77
F6. Right after ACE theft, Defendants had to install Locking Software to avoid conflicts between The Yield Book Online System Downloading ACE Term Structures and The Yield Book Offline System Uploading ACE Term Structures	78
F7. Defendants promptly cleaned off no longer needed Theft Code (B9).....	81
F8. Defendants' admission in the ACE Use Code that The Yield Book was using "multiple sequences <i>fewer</i> than 200 paths", during the period 2000/10-2006/4 when they claim that "their" sequence consisted only <i>one</i> LDS200, besides the phony "1000" paths shows they were using ACE.	82
F9. Defendants' code shows that their new term structure model in The Yield Book was <i>not</i> using their Monte Carlo code, but rather exclusively using ACE for accurate pricing.....	84
F10. Relying on stolen ACE, Defendants for the first time implemented arbitrage trading strategy to cherry pick mispriced MBS/CMO's and hedge them earning profits with virtually no risk and a relatively small amount of capital	85
G. Defendants' Motion to Exclude Mischaracterized the Record and My Testimony	87
H. Defendants withheld The Yield Book production code, TYB-RCS and CMO-RCS, including the offline code read in ACE sequences.....	91

I have been retained by attorneys for Advanced Analytics, Inc. ("AAI" or "Plaintiff") to help to determine if the Defendants have obtained and made use of AAI's ACE sequences and to examine related issues.

1. I am the Frederick L. Moore'18 Professor of Finance and Professor of Statistics, Princeton University. (Primary department: Operations Research and Financial Engineering; Associated departments: Economics; Bendheim Center for Finance; Applied Mathematics).

I have received the following degrees:

Ph. D., Statistics, May, 1989, University of California at Berkeley.

M.A., Statistics, March, 1985, Institute of Applied Mathematics, Academia Sinica, Beijing, China .

B.S., Mathematics, July, 1982, Fudan University, Shanghai, China.

2. I have the following experience in internationally highly regarded professional journals that are directly related to my expert opinions:

Editor, Probability Theory and Related Fields, 2002—2004

Co-Editor, The Annals of Statistics, 2003—2005

Co-Editor, Econometrical Journal, 2007---

Co-Editor, Journal of Econometrics, 2012---

Associate Editor, Econometrica, 2010—

Associate Editor, Journal of Financial Econometrics, 2009-2012.

3. I have received the following honors:

Medallion Lecturer, Institute of Mathematical Statistics, 2011

Fields Institute Distinguished Lecturer in Statistical Sciences, 2010

Guggenheim Fellow, John Simon Guggenheim Memorial Foundation, 2009

Laplace Lecturer, the 7th World Congress on Probability and Statistics, 2008

Morningside Gold Medal for Applied Mathematics, 2007

Humboldt Research Prize, Alexander von Humboldt Foundation, 2006.

Invited lecture (45 min), 2006 International Congress of Mathematicians, Madrid.

President, Institute of Mathematical Statistics, 2006 - 2009

Myrto Lefkopoulou distinguished lecture, Harvard School of Public Health, 2006.

Fellow (elected), American Association for the Advancement of Science, 2005.

Vice Chancellor's Exemplary Teaching Award, Chinese University of Hong Kong, 2002

Faculty Exemplary Teaching Award 2001, Faculty of Science, Chinese University of Hong Kong.

The 2000 Presidents' Award, Committee Of Presidents of Statistical Societies (COPSS)

(American Statistical Association, International Biometric Society --- Eastern North American Region and Western North American Region, Institute of Mathematical Statistics and Statistical Society of Canada)

Fellow (elected), American Statistical Association, 1999.

Fellow (elected), Institute of Mathematical Statistics, 1997.

Member (elected), International Statistical Institute, 1996.
Ruth and Phillip Hettleman Prize for Artistic and Scholarly Achievement, University of North Carolina at Chapel Hill, 1996.
National Science Foundation Postdoctoral Fellowship, 1993-96.
Junior Faculty Development Award, University of North Carolina at Chapel Hill, 1990.
Evelyn Fix Memorial Medal, University of California at Berkeley, 1989.
University Fellow, University of California at Berkeley, 1987-88.
Regents' Fellow, University of California at Berkeley, 1986-87.

4. My 2008 Curriculum Vitae, including a partial publication list for the preceding 10 years, was attached to my previous declaration submitted to the Court. My updated full Curriculum Vitae of has been provided to Plaintiff's counsel and is available at the Princeton University website.

Summary of Findings

5. The evidence leads to the inevitable conclusion that Defendants' disclosed practices were a front for the actual use of the ACE and ACE derivative sequences by the Yield Book to generate outputs, e.g. to calculate prices and OAS values of securities. ACE was used by Defendants in two ways. First, Defendants provided Yield Book **internal users**,¹ including their traders, with ACE verbatim to value MBS and other fixed income securities². Second, Defendants relied on ACE to find derivative sequences by targeting ACE and then provided them to Defendants' buy-side customers or other **external users**³ who subscribed to The Yield Book to value securities. The additional speed and accuracy of ACE enabled Defendants for the first time to implement an arbitrage trading strategy to cherry pick mispriced illiquid MBS and hedge them earning profits with virtually no risk and a relatively small amount of capital. Supporting these conclusions are the following findings:

a. The Yield Book code contained hidden code which was plainly designed to steal the ACE sequences ("ACE Theft Code") and which was used during the last ACE test on June 3, 1998 to steal ACE. *See* Section E.

¹ The Yield Book **internal users** are employees of Defendants and their affiliates, including their traders. The non-**internal users** of The Yield Book are **external users**.

² MBS are mortgage backed securities, securities backed by a pool of mortgages. The MBS varieties include, but are not limited to, pass-throughs (which distribute their cash flow pro rata by ownership interest in the security), IO's (interest only "strips", which receive all the interest paid from a pool) and PO's (principal only "strips", which receive all the principal paid from a pool), ARMs (adjustable rate mortgages) and CMOs (collateralized mortgage obligations which distribute principal and interest to various "tranches" (A, B, C etc. tranche) in accordance with a formula unique to each CMO.

³ See fn 1.

b. Then, right after the ACE theft and in unexpected places, Defendants installed code to surreptitiously allow their **internal users** to use the stolen ACE verbatim in production ("ACE Use Code"). The ACE Use Code was designed to permit two types of operations by The Yield Book. Every day, one type of the code ("**offline code**") surreptitiously read in ACE to generate interest rate paths and stored them in a hidden directory in Defendants. The other type of the code ("**online code**") allowed the interest rate paths generated by The Yield Book term structure model⁴ code produced in discovery to be replaced with the ACE Interest Rate Paths retrieved from the hidden directory. If the **online code** does not find any ACE Interest Rate Paths in a hidden directory, then it uses the sequences generated by The Yield Book code produced, sequences the evidence shows were the ACE derivative sequences. REDACTED

REDACTED which were not as accurate as the ACE sequences. This feature allowed the **online code** to be used in two ways, first, to provide **internal users** with the benefits of ACE verbatim and second, to provide **external users** with use of the less accurate ACE derivative sequences. This scheme facilitated Defendants' arbitrage trading strategy as it made more relatively mispriced MBS available on the market for ACE to identify. The **offline code** is called "offline" because most of the **offline code** was encapsulated under the restriction REDACTED meaning it was not accessible by just anyone with access The Yield Book code. It is accessible only by Defendants' those that meet this definition, the "**inner circle**"⁶. The **online code** was accessible to other persons using the Yield Book code itself, persons indicated by the restriction REDACTED This would likely include The Yield Book, Inc. employees who were responsible for making The Yield Book programs available to the **internal users** and **external users** to value securities. For example, The Yield Book group employees who worked for The Yield Book, Inc. and had access to The Yield Book RCS archive files would be this type of Yield Book user. These Yield Book users could access the **online code**. See Section F.

c. The **offline code** generated interest rate paths using ACE. Comments in the code (Sections C) and requirement in the code (Section F8-9) preclude the possibility that Defendants could be using the sequences that they claim to be using (2nd interrogatory responses, Ex. W) much less the sequences they produced (Section B). Further, Defendants' tests of ACE show it was at least an order of magnitude more accurate, not just much faster, than their production sequence in use at the time of testing (Section A) and Defendants had tested or considered no other sequences on their premises when they tested and stole ACE (Section C and E). Right after the theft of ACE, changes were made to the ACE Theft Code, which was **offline code** (subject to the restriction REDACTED to convert it to be part of the ACE Use Code to retrieve and use interest rate paths generated by **offline code** using ACE ("ACE Interest Rate Paths") (Section C and F2).

d. Calibration is the daily process The Yield Book system used to "calibrate", i.e., readjust, the interest rate model parameters so that The Yield Book could price liquid securities to match

⁴ Term structure model, i.e., interest rate model, uses a sequence to generate a set of interest rate paths that can be used to price fixed income securities and illiquid MBS. See Fig.1 in Exhibit SSS for a visual explanation of where the term structure model part of The Yield Book program fits into the valuation process.

⁵ Literally, REDACTED

⁶ This **inner circle** had access to the stolen ACE sequences.

their observed market prices. Code produced by Defendants indicates that Defendants relied on a combination of ACE sequences – a Super ACE sequence - exclusively for the purpose of calibrating of the term structure model for their **internal users** who then used ACE sequences that are sub-sequences of this Super ACE sequence, to price securities. See Sections C and F.

e. Defendants have not produced the sequences used by The Yield Book as directed by the Court Orders. In fact, *none* of the sequences produced by Defendants in discovery could actually be generated by The Yield Book. Further, Defendants have produced neither a single test system nor a complete set of code files to compile a test system to test a sequence. Defendants produced no genuine development files for LDS200, and no development files at all for LDS100, or either of the two purported 1000 path sequences. The default seeds in the RCS code produced purportedly for LDS100 and LDS200, are unreliable and unverifiable. See Sections B and H.

f. Defendants massively tampered with the test results produced to Plaintiff both right after the last test of ACE in June 1998 (when the evidence shows it was stolen) and then again in response to discovery requests in this litigation. Defendants did so to mask the fact that ACE was far superior to the Defendants' sequence used in production. The expert reports of Dr. Michael Johannes and Dr. Anthony B. Sanders, both dated June 18, 2012, (Ex.s⁷ C and D), failed to rebut any points in the damage theory that I advanced in my Supplemental Expert Report (Ex. B) concerning the value of ACE in profitable MBS trading, and relied on tampered with ACE test results.

g. Defendants produced no files showing independent development of any sequence allegedly developed after the last test of ACE in 1998 and, on top of that, during the course of discovery Defendants destroyed the key evidentiary files, including all original code files relating to the seed selection algorithm. The only files they produced that purportedly show any development of any sequence -- the alleged LDS200 -- had their "date modified" record altered in such a way as to mask the fabrication of the files made after the alleged development of LDS200 almost 5 years earlier, in September 2000. It appears that Defendants had fabricated or altered the files in 2005, expressly for the purpose of making the misleading and, indeed, fraudulent production (contained in CGM 07077 – CGM 07081). The expert report of Dr. Nathaniel Polish dated June 18, 2012 ("Second Polish Expert Report") (Ex. NN) concerning the Defendants' alleged sequence development and testing files, does not support at all that Defendants undertook any genuine development effort in accordance with what Dr. Teytel, Defendants' researcher, contended was his Mixed Seed Algorithm (CGM 00178-179)(Ex. EEEE). Dr. Polish admitted that he never actually ran any The Yield Book code to verify the purported test outputs purportedly used by Dr. Teytel. The data Dr. Polish relied on to draw his conclusions are unreliable, unverifiable and, in fact, fabricated, see D6. Further, my 2nd Declaration dated April 24, 2008, ("2nd Declaration") ¶¶ 56-60 (Ex. E), showed that Teytel's Mixed Seed Algorithm had no mathematical validity for very specific reasons, and that the real algorithm that selected all LDS sequences targets ACE. Defendants and their experts never even attempted to rebut my analysis with a detailed and well-reasoned analysis. Tellingly, the Fifth

⁷ Unless otherwise indicated, all references to "Ex. _" are references to Exhibits attached to the Declaration of Kimberly Walker, dated June 6, 2013.

Declaration of Nathaniel Polish dated May 15, 2008, (“Fifth Polish Declaration”, Ex. F) was submitted specifically in reply to my 2nd Declaration; yet it fails to provide any detailed and well-reasoned rebuttal. *See* Section D.

h. When Defendants stole AAI’s ACE sequences (Section E below) in June 1998 and thereafter installed them into The Yield Book production code (Section F below), they simultaneously installed code to provide The Yield Book with the hedging capability, a capability that was necessary to follow an MBS arbitrage strategy to earn riskless profits. The fact that they are still hedging today supports the conclusion that ACE was the game changer that allowed them to earn these riskless arbitrage profits. *See* Sections C and F10 below. This is exactly the type of strategy discussed in my Supplemental Expert Report dated May 17, 2012, Ex. B. Defendants’ expert reports never addressed the use of this hedging capability for arbitrage. Apparently, Defendants never told them about it.

i. Defendants removed from their production of The Yield Book group’s RCS code (“TYB-RCS Code”, CGM0177) *all* The Yield Book production code relating to generating interest rate paths and produced only portions of the Mortgage Research group’s RCS code (“MR-RCS Code”), *see* D4 and H, and only for the period before July 2005. Defendants also withheld all the production code from their CMO Analytics Group (“CMO-RCS”) as disclosed by Defendants on July 13, 2012. They, of course, also withheld the small, but inculpatory, **offline code** reading in ACE. Defendants have produced no test systems for any sequences, *see* Section H.

Top Level Description of ACE Theft Code and ACE Use Code Flow Chart

6. Defendants’ produced approximately 3,400 RCS code files which are extremely difficult to analyze, not just because of large numbers of interconnected files and functions, but because of the large number of missing The Yield Book code files. The missing files prevented us from compiling Defendants code to building any The Yield Book applications. The ability to compile and build an application is necessary to use normal code debugger tools to analyze this scrambled spaghetti code. Plaintiff asked for the missing files but Defendants refused to produce them and the Court upheld that refusal.

7. Defendants’ fabricated sequences, created by Dr. Radak and produced in 2007 (“Radak Sequences”), and fabricated sequence development files and testing files, produced on March 11, 2011 (“Production”) were of little help to me. Defendants packed the Production with as many as 29 copies of what turned out to be identical RCS code of July 2005 updating previous produced in May 2005. *See* Section D. Nevertheless, after great difficulty, I found The ACE Theft Code (Section E) and ACE Use Code (Section F) tucked away in totally unexpected locations in the RCS code. It is best to describe this code in flow charts attached as Ex. CCC and below.

8. The pricing module of the Yield Book system before ACE tests is illustrated by a simple flow chart, *See* Fig. 1 of Ex. SSS, which showed three components: the Sequence Generator to

generate random numbers (also called “sequence”), the Interest Rate Model (also called the “term structure model”) to generate interest rate paths using the sequence and the Cash Flow Generator to calculate MBS prices, OAS and other outputs.

9. To test ACE, Dr. Wang replaced the REDACTED the left of Fig. 1 by reading in ACE sequences, as displayed by *only the change on the left side* of the flow chart by the dotted lines in Fig. 2, *Id.* Only the REDACTED r was replaced by a function to read in ACE, see a1 in Table A in ¶17 below. The ACE sequences were supplied by Dr. Wang on a floppy disk. In order to avoid detection, Defendants did not install either the ACE Theft Code or the ACE Use Code in any code files relating to generating sequences, nor did they install the ACE Theft and Use Code in any of the other code files designed to implement their REDACTED

Instead, Defendants tucked away the Theft and Use Code elsewhere among the thousands of other RCS code files. The Theft and Use code was implemented later in the process as shown in the right side of Fig. 2 and Fig. 3 by the dotted lines, Ex. SSS.

10. The Defendants claim that (1) they did not save ACE or the interest rates generated by ACE and (2) after the ACE tests, The Yield Book code had no change from pre-ACE tests relating to the sequences used by The Yield Book, other than changes to REDACTED : from a file that uses a single seeds, in file `gauss_random.c`, to one that uses mixed seeds, in file `gauss_random_mixed.c`. See e.g. their interrogatory answers (Ex. W, Nos. 1-4). The code, as explained in Sections E and F, shows that these representations were simply false. First, the RCS files produced contain the ACE Theft Code which was clearly designed to clandestinely save the interest rate paths (term structures) generated by ACE sequences (“ACE Interest Rate Paths”). This Code was used during the 4th ACE test to acquire ACE. That is why Dr. Teytel was able to target ACE while developing his LDS sequences (Section D and Ex. E and G). The ACE Use Code shows that Defendants have been using ACE (Section F) since the theft. From the ACE Theft Code the Defendants were able to save the interest rate paths generates by ACE in a hidden directory, as shown in the flow chart of Fig. 2 of Ex. SSS. This is demonstrated in their code file REDACTED produced in CGM0176 as explained in Section E below.

11. The ACE Theft Code surreptitiously intercepted and “saved” the ACE Interest Rate Paths generated by ACE sequences (Defendants also call the rate paths “term structures”), when the ACE sequences were run through the rigged The Yield Book test system during the last ACE test. Using the saved ACE Interest Rate Paths with their own term structure model and the model input information that Defendants’ saved, the ACE sequences were backed out, one by one. Compare the term structure outputs in the flow charts of The Yield Book *before* preparation for the ACE tests, Fig. 1, in Ex. SSS, to that of the rigged ACE test system used during the last ACE test to steal the sequences, Fig. 2.

12. Then I found that after the ACE theft in June 1998, Defendants immediately installed ACE Use Code to surreptitiously use the stolen ACE sequences in The Yield Book production, See Fig. 3, *Id.*, for a graphical depiction of what happened and Section F below for an explanation of the underlying code used to accomplish this. The ACE Use Code shows that the

ACE sequences are used daily to generate the interest rate paths. Defendants referred the files containing ACE Interest Rate Paths as REDACTED.⁸ The ACE Interest Rate Paths replace the rate paths generated using Defendants' sequence generator every day in a shell game. The Yield Book code file REDACTED performs this shell game of substitution of ACE rate paths generated offline for the rate paths generated online using their own sequence generated online by their sequence generator. For REDACTED the program used to value CMO's, the same substitution function is performed by their code file aptly named REDACTED

13. The ACE Use Code confirms that Defendants began to rely on the stolen ACE sequences for production right after the ACE theft in June 1998, until May 2005 when the RCS code files CGM0176 and CGM0177 were produced. This is shown in the Sequence Chronology in *Fig. 6, Id.* This discovery confirmed Plaintiff's allegations of theft and use of ACE, as shown in *Fig. 5, Id.* and exposed defendants' misrepresentation of the sequences used by The Yield Book as shown in *Fig. 4, Id.* which reflects Defendants' representations of which sequences were used when made in their interrogatory responses (Ex. W, Response 1-4) and 1000 path sequence transmittal letter, p.1 (Ex. Z).

14. Ex. CCC displays an example of three types of code to show some functions in oval figures to illustrate part of the ACE Theft Code and ACE Use Code. An arrow indicates a function call which is an instruction to use another function and the color coded meaning is as follows:

(i) Functions in white; those were normal functions of The Yield Book that existed pre-ACE testing (and represented by Defendants to be the same post-ACE testing), see *Fig. 1, of Ex. SSS* for an overview of the process;

(ii) Functions in brown; those are part of the functions in the ACE Theft Code, see *Fig. 2, Id.* that starting on August 1, 1996, began to be added to the ACE Theft Code. These functions were revised many times after that in lock step with the progress of the ACE tests during almost two years according to the log entries. The Theft Code was finally installed in the ACE test system used on June 3, 1998, and used to steal ACE. The Theft Code was then upgraded to become part of the ACE Use Code right after the ACE theft;

(iii) Functions in grey; those are part of ACE Use Code that was installed right after the theft of ACE to provide Defendants with the ability to use the stolen ACE, see *Fig. 3, Id.* It might have taken one to two weeks to back out the ACE sequences from the saved ACE term structures. The ACE Use Code, including code files containing REDACTED and REDACTED, were created and installed less than 2 weeks after the ACE theft on June 3, 1998. After that, a whole new suite of ACE Use Code was installed during the following month, including REDACTED: for trading and hedging in global financial markets.

⁸ Logically "x" for Dr. Xiaolu Wang who created the stolen ACE sequences.

15. The ACE Use Code was meticulously designed and it is very sophisticated. Even Defendants' The Yield Book **internal users**, who are the employees of Defendants and affiliates, are only provided access to only **REDACTED** which holds the ACE term structures that have been retrieved by The Yield Book **online code**. The **internal users** have no access **REDACTED** the parameters of the term structure model and market curve information that were used to produce the ACE term structures. This limitation prevents them from being able to back out the ACE sequences, like Defendants' inner circle who had access to both did with the ACE Theft Code. Only the **inner circle** employees, such as Robert Russell who the log files show wrote most of the Theft Code, would run the **offline code**. The **inner circle**, as discussed earlier denoted by the access restriction **REDACTED** could access both the **REDACTED** which, as discussed in E below, allows the ACE sequences to be backed out. The **offline code** generates ACE term structures using ACE surreptitiously and stores them in **REDACTED** for the **online code** to retrieve. The Yield Book users, as discussed earlier denoted by the access restriction **REDACTED**, who are all The Yield Book users who have access to neither **REDACTED** The Yield Book servers used by **external users** do not store the ACE term structures, so their **online code**, looking for but not finding them, defaults to use **REDACTED** which uses ACE derivative sequences, LDS, for pricing when they fail to get the ACE Interest Rate Paths in the ^{re}See Ex.CCC, **REDACTED**

16. As illustrated by Ex. CCC, Yield Book **internal users** have access to their internal servers, which have ACE term structures installed for use. The Yield Book **external users** may have access only to the servers, which have not been installed ACE term structures. Then the function `pathgen_cover()` will call the function `pathgen()` and use "Defendants' sequences" such as the ACE derivative sequences, LDS sequences to generate interest rate paths for use.

17. The similarities in the three steps in the ACE Theft Code (in the ACE test system), shown in Table A, to the three steps in the ACE Use Code shown in Table B are evident:

Table A.

ACE Theft Code - on June 3, 1998, used three steps to steal ACE, see Fig. 2, Ex. SSS:

ACE test code written by Dr. Wang (ACE Test System was not produced).	(a1) Purpose: Read in ACE into TYB Interest Rate Model code to generate ACE Interest Rate Paths;
The functions (a2) and (a3) below describes additional ACE Theft Code that was hidden from Dr. Wang and illustrated by the broken lines in the right side of Fig. 2.	
(a2) "Save" the ACE rate paths into hidden files.	
(a3) "Get" the saved ACE rate paths to back out the ACE sequence from the hidden files.	

Table B.

ACE Use Code, right after June 3, 1998, the 3 steps to use the stolen ACE sequences, see Fig. 3, Ex. SSS:

Small offline code withheld by Defendants	(b1) Read in ACE into TYB term structure model to generate ACE rate paths, same as (a1), but may use environmental variable;
Large offline code not produced, identical function to online code produced in MR-RCS:	(b2) “store” the ACE Interest Rate Paths into hidden files; same as (a2).

The **offline code** above is hidden from the TYB online users, as illustrated by the broken lines in Fig. 3.

online code produced in MR-RCS:	(b3) “retrieve” the “stored” ACE rate paths from the hidden files for TYB internal user production, same as (a3).
--	---

18. It is instructive to compare the Theft Code, as shown in Table A and Fig. 2, with the Use Code, as shown in Table B and Fig. 3, by comparing the 3 pairs of functions (a1) with (b1), (a2) with (b2) and (a3) with (b3).

ACE Use Code functions (b2) and (b3) were minor variations from the corresponding ACE Theft Code functions (a2) and (a3) respectively. ACE Use Code installed functions which resolved the potential conflicts arising in production resulted from the **inner circle**/TYB administrators putting ACE rate paths into ^{REDACTED} for use by **internal users** (Defendants’ traders) for whom the ACE Use Code simultaneously retrieves the ACE Interest Rate Path from the same ^{REDACTED}. These functions were integrated into the chassis code designed specifically for the use of ACE in production and are found in the **online code** in the RCS files produced from The Yield Book group employees (“TYB-RCS”) who make The Yield Book available to **internal users** and **external users** to value securities.

ACE Use Code function (b1) can also be derived from ACE test code function (a1). Both of these codes was withheld by Defendants.

In the ACE Theft Code, the code to ^{REDACTED} the ACE rate paths (a2) and to ^{REDACTED} the ACE rate paths (a3) was concealed from Dr. Wang during ACE tests, and hidden under ^{REDACTED} literally, in the code ^{REDACTED}.

REDACTED In the ACE Use Code, the code reading in ACE to generate ACE rate paths (b1), and to the ACE rate paths (b2) was run in the **offline code** and concealed from the users with access to only the **online code**. The code (b1) is withheld by Defendants in litigation, along with all production code for generating term structure rates in TYB-RCS, see H.

A. Defendants tampered with the ACE test result productions

19. I provided the NOTICE OF EXPERT TESTIMONY OF JIANQING FAN IN RESPONSE TO NOTICE OF EXPERT TESTIMONY OF MICHAEL JOHANNES, Ph.D., ANTHONY B. SANDERS, Ph.D. AND NATHANIEL POLISH, Ph.D. ("Fan Notice"), on July 17, 2012.

20. My background and qualifications are outlined in the Fan Notice which is attached hereto as Ex. A. I repeat and reaffirm herein all the findings in the Fan Notice, as though it were fully incorporated in and made a part of this Declaration.

21. The Fan Notice simply verifies and expands on the findings in my original Expert Report (Ex. G), in particular that their Production200 sequence in use when they were testing ACE was biased. My Expert Report shows clearly that the ACE64 production errors are miniscule compared to Defendants' Production200. The Fan Notice shows that Dr. Johannes' conclusion from re-analysis of the same data was erroneous, even if one ignores the tampering with the data, and relied on data massively tampered with by Defendants. See Paragraphs 2-26, Fan Notice. The fact that Defendants would risk massively tampered with the ACE test results produced pre-litigation and during the discovery process supports the conclusion that ACE accuracy was the game changer that allowed Defendants for the first time to earn riskless arbitrage profits. See Sections C and F. The Fan Notice, at ¶¶ 31-49, also refuted the Second Polish Expert Report which raised completely new issues concerning the Sequence Development File and Testing File Production ("Production") that has never been raised in any of previous expert reports. My original Expert Report could not have properly addressed these issues until Drs. Teytel and Radak were deposed about their sequence development and testing efforts in August 2012, a month after the additional expert reports were due in July 2012, to see what, if any, their explanations were and whether they would impact my conclusions. See, e.g., Section E below.

22. When the evidence of the tampering of raw data was discovered, I displayed it in Excel charts. Viewing these charts, a lay person can easily recognize the tampering by comparing the conflicting raw data produced from the identical tests of the same instruments. For example, with respect to the third test of ACE (which occurred in March 1998), the test results that Defendants produced through the discovery process differed from the test results that Defendants supplied to Dr. Wang in 1998 soon after the third test. A comparison of the two sets of reported OAS test results -- for the same instruments valued during the same 3rd ACE test -- is shown in ¶ 18 of the Fan Notice (Ex. A). See Ex. G of the Fan Notice, charts for the tampered-with data. A *partial* correction of the tampered-with test data, is displayed in Ex. H of the Fan Notice. A lay person can confirm the difference just by comparing the two numbers and will immediately understand that Defendants falsified the ACE test results. Compare Ex. G and Ex.

H of Fan Notice to see the effects of even a partial correction of the tampered with numbers. All the Excel charts attached to the Fan Notice were drawn using Microsoft Excel's standard charting utility functions when the scaled test data was loaded into Microsoft Excel. The test data I loaded were all provided by Defendants.

23. After the tampered-with data is displayed in charts, lay jurors can confirm the obvious tampering. Lay jurors should be able to recognize that with respect to the REDACTED⁹, the errors purportedly shown for the ACE sequence, was actually the error of Defendants' Production200 sequence at the REDACTED. One can see this by looking at the errors of Production200 at the REDACTED. The purported "ACE REDACTED" looks unmistakably similar to the Production200 REDACTED and also unmistakably similar to the Production200 REDACTED for the same instruments, See ¶¶ 22-26 and Ex. E of the Fan Notice (Ex. A). Given that the sample size is only two sequences, ACE and Defendants Production200, the probability of this being a coincidence is effectively zero. Thus the conclusion is unavoidable that Defendants presented the larger Production200 errors of REDACTED wholesale as ACE errors to make the Production200 look better and ACE worse.

24. Understanding other tampering of ACE test results requires an understanding of the basic relationship between the yields and the prices of a bond. It also requires an understanding that interest-only MBS ("IO's") and principal-only MBS ("PO's) together are equivalent to a pass-through MBS because they distribute all the cash flow, interest (IOs) and principal (POs), from a given pool of mortgages. Turning the foregoing relationship into a formula, $A \text{ (pass-throughs)} = B \text{ (IOs)} + C \text{ (POs)}$ must always be true. So if $A=C=0$, then $B=0$, see ¶ 19 of Fan Notice (Ex. A). Accordingly, my conclusion that these test results as well were tampered with because they did not follow the rule that cash flow from given pool of mortgages for the IO's plus the cash flow from the PO's must equal the cash flow or the pass-through does not require a subjective judgment call at all. It is a mathematical certainty like $2+2=4$.

25. In the Fan Notice, I refrained from discussing other indications of tampering by Defendants. For example, comparison with REDACTED. See Ex. E of the Fan Notice. Ex. A.

26. My brief analysis in the Fan Notice of the huge superiority of ACE compared to Defendants' Production200 sequence was based on my new discovery of the fact that the ACE test results were massively tampered with by Defendants. It supports my opinion set forth in my original Expert Report (Ex. G, p. 19 and Ex. D) that the ACE test results showed that it was far superior to the Production200.

⁹ Where there was a drop in the interest rate from the Base scenario by 2% to see how the tested sequences would behave if interest dropped.

B. None of the sequences produced by Defendants can be generated by The Yield Book

27. During his deposition on August 29, 2012, Dr. Branislav Radak was shown The Yield Book code file ^{RED} for the date REDACTED, Ex. Radak 7, Ex. H hereto. He confirmed that there is a function in REDACTED, to REDACTED
Radak 8/29/12 Dep. 159:6-162:13 (Ex. I). When the REDACTED
REDACTED

REDACTED So it would have been easy for Defendants to run The Yield Book to save and produce any sequences they used. Defendants resisted the production of the sequences from 2005 through 2007 on the basis of burden but clearly there was no burden.

28. In an effort to undermine this conclusion and to justify their need to create code so they could generate arbitrary sequences instead of using the original The Yield Book code itself, during the September 5, 2012 hearing, Defendants stated, "It [The Yield Book] REDACTED

See Tr. 39:21-41:3 (emphasis added). Defendants' statement is untrue. *First*, Dr. Radak did not just "write a line of code". He created a new set of code files. *Second*, Dr. Radak's testimony does not support Defendants' representation to the Court. Dr. Radak testified that he *did not know* whether sequences would be saved "before or after variance reduction." Radak Dep. August 29, 2012, 158:6-163:21, Ex. I. *Third*, contrary to Defendants' claim, the REDACTED
REDACTED REDACTED

REDACTED
Therefore, Defendants could have easily run The Yield Book and saved all the sequences which Defendants claimed that The Yield Book generated.¹⁰ There was no need for "Dr. Radak...to write a line of code....", let alone to create an entire set of new code files.

29. The RCS archive files produced by Defendants contain The Yield Book code with the following seeds for generating sequences: a) two sets of default mixed seeds for the two purported LDS100 and LDS200 sequences and b) one seed, REDACTED, for the Production200, the single-seed sequence of 200 paths that Defendants used before they allegedly stole ACE. Other files created by Dr. Radak contain another seed for the generation of the 1000 path sequence installed by Dr. Radak in The Yield Book code in April 2006 ("Radak 1000 Path Sequence"). Defendants claimed that the original Production200 seed, REDACTED
purportedly developed by Robert Russell and used for production

¹⁰ For The Yield Book code extracted from the RCS archive files produced, Defendants could have easily deleted (or "comment out") perhaps just one command line to build, then run The Yield Book, and saved the sequences that Defendants claimed that they used for that date. "RCS" stands for files created by a program called Revision Control System to "archive" in one file all versions of a particular code file. See the Official RCS Homepage at <http://www.cs.purdue.edu/homes/trinkle/RCS/>.

from August 1999 to April 2006 (“Russell 1000 Path Sequence”). We will see that the Russell 1000 Path Sequence is just as bogus as their LDS200 generated with purported LDS100 seeds produced in July 16, 2007 which, after Plaintiff complained about it to the Court, even Defendants were forced to admit was not genuine.¹¹

30. One seed, or one set of seeds, could have not been used by The Yield Book to generate two different sequences with the large number of dimensions used in production. As Dr. Radak testified, this is because **REDACTED** Thus,

REDACTED

Radak Dep. 2007, 38:19-39:19. Ex. K. Accordingly, if one were truly developing a 1000 path sequence, one would expect that a seed found to be optimal at 200 paths would not be a likely candidate. Not surprisingly, Defendants produced no tests showing any testing of the purported Russell 1000 Path Sequence, much less against the Production200.

31. Defendants denied Plaintiff’s allegations that The Yield Book used the **REDACTED** to run real LDS100 and LDS200 sequences that are ACE derivative sequences¹², and that Defendants withheld The Yield Book code that used the stolen ACE sequences verbatim. If Defendants’ denials were true, then The Yield Book has used no more than 5 sequences from 1994 through today (including the nonexistent Russell 1000 Path Sequence, which was only a code name in Defendants’ code referring to a Super ACE sequence). These 5 sequences could be generated and saved by The Yield Book using files from only 3 dates: one date before June 1998 for the single seed sequence of 200 paths; one date in the period from October 1999 to April 2000, for the LDS100 (and the nonexistent Russell 1000 Path Sequence); and one date post April 2000 for LDS200 and the Radak 1000 Path Sequence.

32. The Court’s July 3, 2007 Order (Dkt 75) directed Defendants to produce all the sequences *used* by Defendants after April 1999. Defendants claimed that they directed Dr. Radak to produce the sequences. Nonetheless, Dr. Radak testified on September 28, 2007 and again August 29, 2012 that he has no personal knowledge about what sequences Defendants actually used. He was not hired by Defendants until 2004. Ex. L. Clearly Dr. Radak could not have produced ACE sequences and ACE derivative sequences if Defendants used them surreptitiously in the **offline code** or the used the **REDACTED** (because he had no access to The Yield Book, Inc. RCS as shown below).

33. With respect to the July 3, 2007 Order, not only did Defendants fail to have

¹¹ “However, upon review of the version of this sequence produced on July 16, Defendants discovered that they” produced a 200 path mixed seed sequences generated by the seeds purportedly used to generate their alleged 100 path mixed seed sequence. August 17, 2007 letter from Christopher Moore to Hon. Henry Pitman at 5 (Ex. J).

¹² The Yield Book code contains an

REDACTED

Defendants produced no authentic evidence demonstrating that the default seeds were used, and produced no test system at all to allow others to verify the default seeds. So the default seeds are unreliable or unverifiable. The complete The Yield Book code would need to be produced to verify them.

someone with first-hand knowledge of what sequences were actually used by The Yield Book make the production, but also, the person who made the production, Dr. Radak, admitted that he did not even run The Yield Book and use the inherent function to save the sequences *generated* by The Yield Book. See paras. 29 and 30 above. If he had done so, Defendants would have produced a grand total of three sequences, a single seed 200 and a mixed seed 100 and 200 path sequence for the period after April 1999 and before March 2006. This evidence would have exposed the misleading nature of Defendants earlier argument to the Court on March 1, 2006 that to produce all the sequences would be burdensome because “The random numbers [the sequence] themselves are not saved on a daily basis by Defendants.”, *see* the March 1, 2006 Hearing Transcript, 12:8-21. “We don’t save what two plus three equals on a daily basis”. *Id.* 17:25-18:1. Defendants claimed: “As for burden, we have confirmed that Defendants’ employees would require months of uninterrupted work to identify every immaterial difference among sequence versions after September 1, 2000, and then reconstruct each of those versions.” Defendants’ July 16, 2007, letter to Magistrate Judge Pitman, Ex. M. We now know that was a huge exaggeration because based on their allegations of which seeds they used to generate sequences, there could have been only three sequences, (or four if one included Russell 1000 Path Sequence which was completely fabricated as shown in C below). Moreover, the production did not include the ACE sequences which were surreptitiously stolen and used by Defendants as shown in E and F below.

34. After being ordered on July 3, 2007 (Dkt. 75) to produce the sequences, instead of using The Yield Book to

REDACTED

in the manner it could have been used as discussed in paras. 29 and 30 above, Dr. Radak created a new set of code files (“Radak Code”) to generate new sequences which were produced on July 16, 2007 (CGM04943). Under the guidance of Mr. Robert Russell, who was in charge of testing ACE and began adding ACE Theft Code on August 1, 1996, the day before Defendants contacted Dr. Wang to request an opportunity to test ACE as shown in Section E below, Dr. Radak inserted code commands¹³ to create sequences with bogus dimensions¹⁴, bogus path numbers and bogus seeds that permitted them to manufacture the sequences Defendants produced. The July 16, 2007 production (CGM04943) has

REDACTED

All the sequences were generated from the **REDACTED** all with phony dimension numbers and/or the wrong number of paths, or wrong seeds. Even this fabrication demonstrates that the sequences did not change on a “daily basis” as Defendants implied at the March 1, 2006 hearing.

35. In response to Plaintiff’s August 3, 2007 letter to Magistrate Judge Pitman pointing out that Defendants’ July 16, 2007 production (CGM04943) contained redundant and phony sequences, Defendants admitted that they produced a phony LDS200 sequence using the purported default LDS100 seeds. But on August 17, 2007 they replaced the July 16, 2007 production (CGM04943) by another production created by Radak (CGM04945) who admittedly had no personal knowledge of what sequences were actually used. This production again

¹³ Such as the function **REDACTED** to generate **REDACTED**

See

REDACTED

created by Radak CGM04947 Ex. N.

¹⁴ As shown below, the number of dimension generated the code produced is

REDACTED

as in the

contained all phony sequences. Responding to the Court's August 31, 2007 Order (Dkt. 81) to produce the 1000 path sequence that had been only recently disclosed, Defendants produced two purported 1000 paths sequences on September 7, 2007 (CGM04947) which, as discussed below in Section C below, were created to cover up their use of ACE.

36. The Court Order of August 31, 2007 (Dkt. 81) explicitly directed Defendants to provide a declaration by a person(s) with first-hand knowledge attesting to the authenticity of the sequences produced by Defendants. In response, Defendants provided the declaration of Dr. Radak, dated September 1, 2007 (Ex. O), who, at a later deposition, admitted that he had no first-hand knowledge: "Q. Well, is the -- is the code that generates a sequence in ... the yield book RCS? A. Yes. Q. And did you check the yield book RCS for the code that they used to generate sequences? A. I don't have access to yield book RCS." Radak September 28, 2007, Dep. 11:25-12:5(Ex. L). Further, Dr. Radak was not hired by Defendants until 2004 (Radak September 28, 2007, Dep. 9:16-17, Ex. L) and therefore could have had no first-hand knowledge of the sequences used before 2004, no matter what code he had access to. With no access to The Yield Book, Inc.'s files at all, he could not possibly have had first-hand knowledge of whether anyone in The Yield Book group used seed files to generate ACE derivative sequences using the environmental variable "sfile" that Defendants admitted was contained in RCS code. He could not possibly have known whether The Yield Book, Inc. used ACE sequences verbatim in production in **online** or **offline code**, because the archived The Yield Book code was contained in RCS files that existed only in The Yield Book, Inc. which he did not have access to. Defendants appear to have produced some RCS files from The Yield Book, Inc. in their production CGM 00177 but they carefully removed all of the term structure model code which generates interest rate paths using sequences and refused to produce them.

37. With respect to the number of dimensions in the sequences generated by The Yield Book, they all have the ^{REDACTED} 5. In Robert Russell's final deposition, after being shown the RCS code, he recanted his earlier testimony about **REDACTED**

Russell Dep. 375:2-16, Ex. Q. Yet the dimensions of all the Radak Sequences in Defendants' July 16, 2007 and August 17, 2007 productions are ^{REDACTED} as indicated in Dr. Radak's September 1, 2007 Declaration falsely attesting to their authenticity. Ex. O. The two 1000 path sequences created by Radak and produced on September 7, 2007 also had **REDACTED**

38. We shall refer to all the sequences produced by Defendants as "Radak Sequences" because they were not generated by The Yield Book code. Not only are the Radak Sequences not the sequences used by The Yield Book, they cannot even be generated by The Yield Book code. The Radak Sequences were generated by Dr. Radak's custom written Code.

¹⁵ This is indicated by the

REDACTED

Robert Russell dep. 373:12-21 Ex. Q **REDACTED**

If you had a tape measure 120 inches (10 feet)

long, **REDACTED**

39. All the single seed sequences produced by the Radak Code, including the one purportedly to be the 200 path sequence used before 1999 and the purported two 1000 path sequences, are of **REDACTED**

Yet according to Defendants, all the sequences generated by The Yield Book before 2007 have **REDACTED**

See, e.g. “A Term Structure Model and Pricing of Fixed-Income Securities, pp.1 and 11, Ex. S.

40. The Radak Code shuffled and mixed up the two factors for each sequence. The Radak Code named the variables storing all the sequences Defendants produced “zz_dump”, and the purpose “dump_random_numbers”, *see monte_thousand_19990801_86int.c*, Ex. N. Analyzing Radak Sequences in such “dump”, since the Radak Code shuffled and mixed up the two factors, it is garbage-in and garbage-out; at best, it is totally meaningless, and at worst, misleading, like Dr. Polish’s purported comparison of ACE with these Radak Sequences. Dr. Polish expressly stated that he relied on Dr. Radak’s September 1, 2007 declaration, Ex. O, which, as shown above, was not based on personal knowledge of the sequences used by Defendants, for the proposition that the Radak sequences were in fact the sequences used by The Yield Book. See Exhibit 1, p.4 to Dr. Polish’s 4th declaration dated February 15, 2008, submitted in support of Defendants summary judgment motion. *See ¶¶ 2-5, Ex. YYYY.*

41. In all Defendants’ productions, in all the sequences in each production, Defendants represented that the **REDACTED**

42. For 1994-10-01, **REDACTED**

The abbreviated notation, *e.g.*, **REDACTED**

other sequences, the mixed seed sequences, just haven’t been

REDACTED

¹⁶ The

¹⁶ See Code in **REDACTED**

REDACTED so the error is obvious. But **REDACTED** indeed they are.

43. If one extracts a copy of The Yield Book code file **REDACTED** from the time frame of the purported Russell 1000 Path Sequence from the RCS version **REDACTED** produced, this code file will have approximately **REDACTED**. The Radak Code performing the analogous function to **REDACTED** is the code file **REDACTED**. Yet it has only **REDACTED**.

44. The RCS code files are supposed to be The Yield Book code files archived by Defendants to keep track the revisions over time, using a public tool known as Revision Control System. The partial RCS archive produced from Mortgage Research has over **REDACTED** code files. Defendants inexplicably removed all the code files of term structure model for generating interest rate paths from The Yield Book, Inc. RCS files produced. Defendants disclosed only on July 13, 2012 (Ex. ZZZZ) that they also have withheld all RCS files from CMO Analytics Group, which is an integral part of The Yield Book and allegedly used by Dr. Tetel to develop the LDS200 sequence. Altogether, there are **REDACTED** of RCS code files. If the complete and up-to-date RCS code files were produced, for any given date one could extract The Yield Book code for that date. Then one can compile and link these code files to build The Yield Book application for that date. One can then generate the sequences that actually can be generated by the default seeds, although not by the real seeds stored in *sfiles*, the files used by the environmental variable to generate a sequence from a file of seeds.

45. Clearly, the Radak Code is not part of any RCS code. It is not even portions of any The Yield Book code, nor portions of any extractions of RCS code. The Radak Code in each of the 15 folders named by a date in the July 16 and August 17, 2007 sequence productions (CGM04943 and CGM04945) contains essentially only three C source code files¹⁷: modified *gauss_random_mixed.c* if the date is after 1999 (or *gauss_random.c* if before 1999), *orthovect.c*, which is a generic **REDACTED** code file used everywhere, and a critical custom code file created by Radak with a name such as *monte_thousand_19990801_86int.c*. The Radak Code contains only about a dozen of “head” files. The Yield Book code contains hundreds of “head” files. When it is compiled and run, Radak Code only produces Radak Sequences, not any sequences generated by The Yield Book. Contrary to the Magistrate Judge Pitman’s belief based on Defendants’ misrepresentation, the Radak Code does *not* generate any interest rate scenarios. To do that, one needs a complete set of The Yield Book code which Defendants have refused to produce.

46. On August 31, 2007 The Court ordered (Dkt. 81) that “defendant is directed to produce an affidavit or affirmation pursuant to 28 USC 1746 from individual or individuals with first-hand knowledge attesting to the authenticity of the sequences that have been produced”. My understanding of the import of the language “the authenticity of the sequences that have been produced” is that it required verification that sequences produced were, in fact, the sequences Defendants actually used. It would have been quite simple for Defendants to cause someone who

¹⁷ The only exception is that there is an *unused* The Yield Book code *Monte.c* in the 8/17/2007 production in file 2000-11-01.

worked in the actual The Yield Book group with first-hand knowledge of the sequences actually used to have submitted such a declaration attesting to the authenticity of the sequence produced if, in fact, they were authentic. Instead, they submitted a declaration by someone with no first-hand knowledge, Dr. Radak. The fact that Defendants disobeyed a discovery order requiring them to do so is additional strong evidence that Defendants are really using ACE and ACE derivative sequences.

47. The Radak Declaration (Ex. O) submitted by Defendants on September 2007 does not satisfy the requirement, because Radak admitted that he lacked first-hand knowledge to the sequences used by Defendants. Radak Declaration does not say if Radak has personal knowledge. The Radak declaration stated that “I am familiar with what are known as the Yield book sequences” but avoided stating the critical definition of what he meant by “the Yield Book sequences” or stating if these sequences have anything to do with the sequences used by Defendants. It omitted any reference to the 1000 path sequences. Defendants’ May 18, 2012 letter to Plaintiff (Ex. OO) at 12, fn.6 tried to explain this failure away by claiming:

Prior to Dr. Radak's declaration, Defendants had agreed "with AAI to define the term 'Yield Book sequences' ... to include all sequences used in connection with a Monte Carlo simulation with 200 or fewer paths and available to users of the Yield Book, whether used internally by Defendants or externally." Defendants' July 16, 2007 Letter at 1-2.

48. However, the claim in Defendant’s July 16, 2007 letter that Plaintiff had agreed to production of “200 or few paths” is itself demonstrably false. In fact, Plaintiff expressly rejected such limitation when proposed. See Christopher Moore July 11, 2007 Letter to Alan Loewinsohn (Ex. X) and Alan Loewinsohn’s July 13, 2007 letter response to Moore (Ex. Y). Ultimately, Magistrate Judge Pitman found there was no such agreement when he ordered production of the 1000 path sequence (Dkt. 81). Defendants never submitted a sworn statement by a person with first-hand knowledge as to the authenticity of the 1000 path sequences produced either. Defense counsel’s letters explaining that the sequences produced are the sequences that were used and when they were used (*see* Christopher Moore’s letters of July 16, 2007 (Ex. M), August 17, 2007 (Ex. J), and September 7, 2007 (Ex. Z)) are not based on his personal knowledge and are contradicted by the mountain of evidence presented in this declaration. Thus I cannot put much weight on defense counsel’s explanation in my investigation into what sequences were used by Defendants.

49. In fact, it is apparent that Defendants *excluded* the ACE sequences *in their definition* of the “Yield Book sequences”, see Defendants’ 56.1 Statements, ¶ 108 (Ex. _PP) citing the 1st Polish Declaration of February 28, 2006 at ¶¶ 2-4 (Ex. QQ). Defendants *defined* the Yield Book sequences to be the sequences generated using the default seeds in The Yield Book code extracted from the RCS code produced in CGM 00176-00177. It does not account for the use of the “sfile” environmental variable to read in sets of other seeds nor does it account for the fact that The Yield Book code cannot generate any ACE sequence (as opposed to an ACE derivative sequence). The computer used to test ACE was only fed the ACE sequences themselves. *Their definition* of “Yield Book sequences” only includes sequence generated by The Yield Book which by definition excludes the ACE sequences themselves even if The Yield Book code used ACE. Defendants’ representation and the confusion engendered by changing the

concepts of “sequences used by The Yield Book” to “Yield Book sequences”, defeated the whole purpose of the declaration directed by the Court’s August 31, 2007 Order (Dkt. 81) which required a sworn statement by a person with first-hand knowledge of the sequences used to permit a determination of whether Defendants used ACE sequences.

50. After Plaintiff’s Objection to the Magistrate Judge’s Denial of Plaintiff’s Application to Compel Production of The Yield Book code and to Compel the Deposition of Robert Russell and Steward Herman (“RCS Objection”) pointed out Defendants’ switch from “all the sequences used by The Yield Book” to “Yield Book sequences” in their purported compliance with the Court Orders, Defendants’ opposition to the RCS Objection switched from referencing “Yield Book sequences” to another undefined term “Defendants’ sequences”.

51. Defendants dodged making a representation about the “sequences *used* by The Yield Book” by referring only to “Yield Book sequences” or “Defendants’ sequences”, i.e., the sequences *generated* by The Yield Book, and by the untrue claim that Defendants had produced the *entire* RCS code. See the misstatement at p. 3 in Lawrence Friedman’s Letter of January 25, 2006, to Court (Ex. RR at p. 2) that “Specifically, on May 26, 2005, Defendants produced their entire production code to AAI, along with any and all changes made to that code during the entire relevant period.” The fact is that Defendants withheld the entire The Yield Book, Inc. RCS production code (“TYB-RCS”) relating to sequences and generating interest rate paths, and also withheld many Mortgage Research RCS code files necessary to compile the code so it would generate a sequence or price a security. As only relatively recently disclosed by Defendants on July 13, 2012, they also withheld the entire CMO-RCS code which is an integral part of The Yield Book code and purportedly was used by Dr. Teytel with the Mortgage Research RCS code to obtain a complete set of source code to price securities during his development of his LDS sequences. See Section D below.

52. The sequence development files and testing files produced by Defendants on March 11, 2011 have been heavily tampered with, see ¶¶ 31-35 of Fan Notice (Ex. A) and Section D below. They provided no verifiable evidence to support the claim that default seeds for either of the LDS100 or LDS200 sequences are genuine, or that any of them were ever tested. They provided *no authentic documents* to support the contention that the purported mixed seed algorithm described by Dr. Teytel was ever used.

53. The most reliable evidence of any authentic portion of any LDS sequence is the first portion of the LDS64 generated using the seed 13812 because it was identified in the Teytel Notebook (CGM00209)(Ex. SS) as the “Best” among hundreds of thousands of LDS candidates tested by Teytel, selected out of a gazillion possible LDS sequences. This “Best” seed portion was the first portion of each and every one of the final LDS64 sequence candidates tested by Teytel. Comparison of this “Best” seed portion of LDS64 to the first portion of ACE64 demonstrated that the reason it was labeled as “Best” was because it was selected by targeting ACE. My mathematical analysis shows with a 99.5% certainty (the probability of such a strong similarity occurred by chance is only 5 in 1000) that the real algorithm used by Teytel to select sequence was targeting ACE as I opined in my first Expert Report and 2nd Declaration, Ex. G and E, and see Section G below. So the most reliable Teytel sequence development evidence indicates that the real algorithm used by Defendants is one that is designed to select LDS mixed

seed sequences by targeting ACE. This method would result in Defendants obtaining a set of seeds to generate a derivative ACE sequence and these seeds could be used by the SFILE environmental variable in code file `gauss_radom_mixed.c` to generate the ACE derivative sequence without a trace of such activity being disclosed by the files produced in discovery.

54. In summary, Defendants produced only Radak Sequences in 2007 and the Radak Declaration in purported compliance with the Court Orders (Dckt 75 and 81) directing them to produce the sequences used by Defendants with a sworn statement by persons with personal knowledge attesting their authenticity. Radak's then testified that he has no personal knowledge. The Radak Declaration was so carefully worded that even if the declaration were true, it avoided what it was supposed to do by the Court Order Docket 81, namely, attesting that Defendants produced all the sequences actually used by The Yield Book.

55. As shown above, they have done neither. None of the Radak Sequences can be generated by The Yield Book. Recall that all sequences generated by The Yield Book have ^{REDACTED} dimensions and at least 2 factors while none of the Radak sequences have ^{REDACTED} dimensions, only **REDACTED**, the single-seed sequences have only one factor and all the sequences have shuffled factors. The August 17, 2007 production contains 15 Radak Sequences:

- One of the Radak Sequences of ^{REDACTED} dimensions purportedly used the seed of ^{REDACTED} single seed sequence before 1998;
- Seven of the Radak Sequences purportedly used the purported default seeds of the LDS100 sequence of 1999 and 2000-9-1, 2000-10-1, 2000-11-1; Three (3) of them with ^{REDACTED} dimensions are identical. Four of them with ^{REDACTED} dimensions are identical and are also identical to the first ^{REDACTED} dimensions of the sequence of the purported LDS100 with ^{REDACTED} dimensions;
- Seven of the Radak Sequences purportedly used the purported default seeds of the LDS200 sequence of 2000-10-1, 2000-11-1, 2005-2-1 and 2007-6-29; The first three sequences of ^{REDACTED} dimensions are identical, only the last sequence has minor change and was installed shortly after Plaintiff moved to compel production of the sequences. Three of them with ^{REDACTED} dimensions are identical and are also identical to the first ^{REDACTED} dimensions of the purported LDS200 sequence of ^{REDACTED} dimensions.

56. Using the seeds of only 3 sequences and the Radak Code, Defendants created out of thin air, 15 redundant Radak Sequences that cannot be generated by The Yield Book. I don't see any reason to do that, other than to cover up their prior representation that the sequences generated by The Yield Book changed on "daily basis". This analysis of the phony Radak Sequences generously ignores the phony LDS200 sequences generated by the default seeds of LDS100, and the phony Radak Sequences of dimensions of **REDACTED** in Defendants' July 16, 2007 production which Defendants have withdrawn.

57. The sequence, purportedly of 1000 paths used from August 1999 through April 2006, differs from the other Radak Sequences above in that this sequence does not correspond to any sequence that could have been generated by The Yield Book during that time frame. Rather it corresponds to a Super ACE sequence, as we shall see below.

C. The non-existent Russell 1000 Path Sequence in reality was a code name in Defendants' code referring to a Super ACE sequence

58. The purported Russell 1000 Path Sequence is fundamentally different from the other Radak sequences produced. As discussed above, the other Radak Sequences cannot be generated by The Yield Book, but at least the code can generate sequences with the same number of paths using the same seeds, the default seeds, using The Yield Book code existing during the relevant time period. Thus, sequences with 100 or 200 paths can be generated by The Yield Book. As noted above, the purported Russell 1000 Path Sequence does not even have its own unique seed. Instead it purportedly shares a seed with the single seed sequence of 200 paths. Further evidence that Russell 1000 Path Sequence does not exist at all is that the code produced, and purportedly in use at the time, cannot generate 1000 paths. This code limits the number of paths to a number of 3 digits and "1000" has 4 digits.¹⁸ See Section F7.

59. The purpose of The Yield Book's calculation is to find the market value of relatively illiquid, one of a kind, MBS instruments in terms of liquid instruments currently available in the marketplace. Other than the sequence generator, The Yield Book has two main components, a term structure model which generates a representative sample of future interest rate paths based on the current Treasury yield curve and interest rate caps and swaptions and a cash flow generator. See, Fig.1, Ex. SSS. The set of interest rate paths is then fed into a cash flow generated that calculates the cash flow for each interest rate path for the pool of residential mortgages that backs the illiquid MBS. An error in the interest rate paths reverberates throughout the cash flow generator calculation and the follow on present value of the cash flow calculations because the same set of interest rates are used for both steps in the process. Thus it is imperative that the projected interest rate paths be as accurate as possible.

60. To achieve such accuracy, the term structure model is **calibrated** to current market interest rates and volatility using Treasury yield curves and interest rate caps and swaptions to determine volatility. Without ACE, Defendants would normally use the same sequence to calibrate the terms structure model as it uses to price MBS. Each sequence path generates one interest rate path. Dr. Radak described this process in connection with his development of his 1000 path sequence. Radak August 29, 2012 Dep. 69:6-71:17, Ex. R. But with ACE32, 64 and 128 sequences, Defendants could create a much more accurate sequence for calibration by combining these tested and proven sequence into a larger sequence of 224 paths and then, because ACE has not been subject to antithetic variance reduction technique which takes one set of paths and turns them into four times as many, use the antithetic process to turn the 224 ACE paths into an 896 path Super ACE sequence that would generate 896 interest rate paths. Super ACE would likely be an order of magnitude more accurate than any subset of its constituent sequences such as ACE32, 64 or 128 because it would be more likely to completely cover the entire future interest rate path probability space by triggering the interest rate options used by Defendants in calibration, *i.e.*, caps and swaptions, which are only triggered when interest rates move by a minimum amount specified in the option. Common sense demonstrates that the more well tested paths you have, the greater the likelihood that the interest rate paths

¹⁸ Defendants start counting their path numbers with the number 1. See Ex. XXXX, the first two pages showing the first 4 paths of two sequences produced by Defendants, the Production200 and Russell 1000.

generated would vary by the minimum amount specified to trigger the option in an accurate way.

61. More accurate calibration and pricing would allow Defendants to A) identify and purchase illiquid MBS that were underpriced in terms of a comparable set of liquid instruments that would generate the same cash flow and, B) at the same time, hedge this purchase by selling (“shorting”) a set of liquid instruments that generate the same cash flow. From this set of transactions, they would realize more from the sale of the shorted instruments than they paid for the underpriced MBS. Thus, they could take \$100 to purchase underpriced illiquid MBS and get it right back plus, say, \$1 by selling liquid MBS for \$101 leaving them with their original \$100 plus an additional \$1 conditional profit. The condition is that the short must be accurate enough so that when the position is unwound, when the MBS is sold and the shorted instruments repurchased with the proceeds, Defendants will be left with their \$1. Assuming that the hedge is accurate enough to preserve the \$1 profit when the position is unwound, Defendants could use the same hundred dollars over and over again to set up hedging transactions limited only by their ability to continue to identify underpriced illiquid MBS thus earning arbitrage profits. It would be a riskless, money generating machine.

62. The evidence indicates that this is exactly what they were doing. Over the 15 year period after they stole ACE during which Defendants traded what likely amounted to trillions of dollars of illiquid MBS, the profits could be sizable. As shown in Section A above, ACE was far more accurate than Defendants production sequence in use when the evidence shows that they stole ACE (as shown in Section E below) and used it (as shown in Section F below). Defendants admit to being market makers in MBS including illiquid MBS like CMOs. Market makers as a matter of course hedge their entire portfolio and adjust the hedge on a daily basis to insulate themselves from market risk. In doing so, using ACE Defendants could cherry pick underpriced illiquid instruments for their market making portfolio continuously realizing arbitrage profits when they sold the underpriced illiquid instruments at or close to its true cash flow market value and unwound the hedge by rebalancing their portfolio by buying back the comparable short positions with the proceeds of the sale the illiquid instruments. Of course they can only do this if they can hedge accurately enough. The code produced shows that they did not begin to allow their traders to hedge until just after they stole ACE demonstrating that ACE was the game changer that allowed them to follow an arbitrage strategy.

63. Defendants’ code did not permit hedging before they stole ACE.¹⁹ Thus, they

¹⁹ Defendants produced computer RCS code files (CGM00176 and 00177) that Defendants contended contained all the versions of the files needed to generate all of the sequences Defendants ever used during the relevant period. See Defendants responses interrogatories 1-4 contained in Plaintiff’s second interrogatories, Exhibit Ex. W. These files show that Defendants used sequences with The Yield Book to value and hedge illiquid MBS. These files show that Defendants did not begin to create the code file later used for hedging, `hedge.c`, until a little more than two weeks before the last test of ACE, when they stole it. Before that, the code shows that Defendants had been thinking about hedging and experimenting with hedging since at least 1995 as shown by a file called `hedging.c` revised in 1995 by Robert Russell, the employee of Defendants in charge of testing ACE and whose initials appear in the RCS log file entries which indicate that he wrote the ACE Theft Code. Section E. Thus, the code produced demonstrates

clearly did not view their then existing single seed 200 path production sequence ("Production200") capable of supporting an arbitrage strategy that required hedging. Using a Super ACE sequence for calibration of the term structure model while using the smaller and quicker ACE constituent sequences for accurate pricing appears to have been the game changer that enabled them to successfully follow this strategy. This is demonstrated by the fact that prior to the theft of ACE, Defendants were only experimenting with hedging as evidenced by the comments in `hedging.c,v` (CGM00176). This code file contains a comment at line 54-55 as follows:

```
hedging.c(*this function is for experimental purposes and is
inactive*).
```

Defendants could not hedge their portfolios. The code in `hedging.c,v` was not sufficient to accomplish it. Defendants' ACE testing showed them that ACE64 was an order of magnitude more accurate than their Production200. Fan Notice ¶ 20 (Ex. A). It appears that during the third test of ACE, Defendants became comfortable with the idea that that ACE was so accurate that they could hedge their portfolios in an arbitrage strategy that they started to create code to allow hedging in production. On March 20, 1998 they created first version of `hedge.c`, version 1.1. This file was modified 10 times in two and a half months prior to the theft of ACE on June 2, 1998, and Defendants continued efforts to refine `hedge.c` were not completed until May 25, 2000. A copy of the RCS version of `hedge.c,v` containing various versions of `hedge.c` and showing the revision dates is Ex. MM.

64. The potential profits to be made from arbitrage enabled by ACE certainly explains Defendants' motivation for the following behavior a) extensive testing ACE over a two year period all the while deceiving Dr. Wang about their true intentions, as shown in Section E, and then stealing it, b) taking the risk of producing phony sequences in this litigation, withholding from production in discovery the code to cover up their use of ACE and failing to provide a sworn statement by a person with knowledge of sequences they actually used in direct violation of this Court's order as shown in Section B, c) falsification of ACE test results both before and after this litigation as shown in Section A, d) production of no or incomplete and tampered with sequence development and sequence testing files while destroying the original files during this litigation as shown in Section D, e) commencing the installation of code to provide a hedge capability to traders in The Yield Book code by adding `hedge.c` just prior to stealing ACE as

that hedging MBS was clearly something Defendants wanted to do when Dr. Wang came along in 1996 with his ACE sequences. So naturally, they were interested in ACE; they just never planned to pay for it. This is demonstrated by the fact that the RCS code files show that Robert Russell began installing code that would allow Defendants to steal ACE the day before they first contacted Dr. Wang, on August 2, 1996, to request the opportunity to test ACE. See Section E below. The RCS code files further show that shortly after the theft of ACE during the last test in early June 1998, The Yield Book code was again altered to toss out the interest rate paths generated internally by the default seeds in The Yield Book code produced and to substitute a set of interest rate paths generated by ACE sequences with code not produced by Defendants. Section F10.

shown in Section C and f) shortly after stealing ACE, implementing the use of the ACE generated interest rate paths in The Yield Book code though their online code changes to use offline ACE generated interest rate paths as shown in this Section F. Failing to produce the necessary profits documents that are required to determine their profits from illiquid MBS before and after the theft of ACE in June 1998, as Defendants have, fits right in with this pattern of obstructive and deceptive behavior.

65. Defendants were well aware of the concept of arbitrage with respect to illiquid MBS when they were testing ACE. They described what it would take to achieve arbitrage free pricing in their June 1997 paper entitled “A Term Structure Model and the Pricing of Fixed-Income Securities”, at 12, Ex. S, which demonstrates that they were well aware of the arbitrage advantage of more accurate pricing that would follow from the use of ACE. The article was co-authored by Robert Russell who was, at the time of publication, in charge of testing ACE and who had already added the first ACE Theft Code components almost a year earlier. See E below. This demonstrates that at no time did Defendants deal in good faith with Dr. Wang. They had planned to steal ACE if it were proven to be sufficiently accurate during testing right from the beginning.

66. The fact that the purported Russell 1000 Path Sequence **REDACTED** alone casts grave suspicion on the authenticity of the Russell 1000 Path Sequence. Dr. Radak testified that he was asked to develop the Radak 1000 path sequence **REDACTED** Radak Dep. August 29, 2012, 51:16-52:5. Ex. S. **REDACTED** Radak Dep. August 29, 2012, 59:14-19. Ex. S. This defies common sense. If the Russell 1000 Path Sequence had actually existed, one could not know whether the replacement sequence was better without testing against **REDACTED** Radak Dep. August 29, 2012, 54:12-16. Ex. S. It would have been easy **REDACTED** But if there was no Russell 1000 Path Sequence, **REDACTED** Radak produced no testing files for his 1000 path sequence. If he had produced the source code he used, or Defendants had produced the updated RCS20 The Yield Book code post 2006, it would show the modification he had to make to generate 1000 paths. This would have exposed the fact that prior to Radak’s changes, the production code could not generate 1000 paths and that the purported Russell 1000 Path Sequence produced was fabricated out of whole cloth for discovery purposes.

67. The Russell “1000” path sequence was apparently fabricated to provide an excuse for certain Yield Book code file references to a purported **REDACTED** But the code limitation demonstrates that the reference cannot be to a **REDACTED** and that it is really a veiled reference to the

²⁰ RCS (Revision Control System) code is an archival code file created using open source RCS program that contain all the versions from ver. 1.1 to the latest version of a particular code file that the programmer saved. The Mortgage Research Group and The Yield Book group each had their own set of RCS files to archive their version of The Yield Book.

use of a Super ACE sequence. As discussed above, calibration is the daily process for setting the parameters of term structure model of The Yield Book so that it can replicate the market prices of liquid instruments like Treasuries, swaps and interest rate options. In order to trigger an option, the interest rates need to move up or down by a certain minimum amount. In addition to being more accurate, the use during calibration of an accurate sequence having more paths has the added advantage of increasing the likelihood of triggering all of the volatility options, caps and swaptions. So a large sequences could be optimal for calibration even it might be too large for normal MBS pricing when surveying the market for underprices MBS as described above. Thus while a Super ACE 896 path sequence would be very desirable for calibration, it would not necessarily be better for pricing illiquid MBS to find underpriced instruments because of the time it would take to price them.

68. Both The Yield Book code files `nts_param_to_mse_option_prices.c` starting with version 1.3 dated October 12, 2001, and `nts_monte_calibration.c` version 1.1 dated April 8, 2002, make reference to using a “1000” interest rate path sequence in calibration. The comments in the code files essentially say that the set of “1000” interest rate paths contains as a subset the smaller set of interest rate paths actually used for pricing.

69. The calibration code `nts_monte_calibration.c,v`, Ex. T, has total of two versions: V1.1 of 2002.04.08, and V1.2 of 2003.01.15 (Ex. U), which were used at least until May 2005. In all versions of `nts_monte_calibration.c`, there are comments under the function `nts_monte_backward_induction()`, *Id.* p.5:

```
/* For option exercise purpose, generate a larger sample of 1000 paths,
** containing as a sub[s]et the given set. The larger
** number is needed for getting prices as a trigger to calls:
** sample->(x,u,v)[i][isample]*/
/* the prices (at time0) returned by this function is however still
** based on the smaller set; so that the smaller set is
** used in the calibration*/ (emphasis added)
```

70. The second half of the above comments is repeated, and also embedded in the second half of `nts_monte_calibration.c`, under the definition of the function `nts_backward_induction()`, Ex. s T and U (at p.8):

```
/* the prices (at time0) returned by this function is however still
** based on the smaller set; so that the smaller set is
** used in the calibration*/
```

In C-code, comments by programmers are intended to be read and not executed by computers are enclosed within `/*...*/`. The `**` at the beginning of each line is for clarity. The above comments are also in all versions of The Yield Book code `nts_param_to_mse_option_prices.c` Ex. V, p.163.

71. As explained above, calibration may require a larger sample of interest rate paths to trigger the option price. So that it is not unexpected that the comments `/* For option exercise purpose, generate a larger sample of 1000 paths`,”.... “The

larger number is needed for getting prices as a trigger to calls: sample->(x,u,v)[i][isample]" explains that the purpose of Defendants' sequence of so called "1000" paths is to cause the triggering "calls" which are one form of option used by Defendants in calibration.

72. One sequence path generates one interest rate path. A "1000" path sequence is needed to generate a set of "1000" interest rate paths used for calibration. If the "1000" interest rate paths contains as a subset, the interest rate paths used by The Yield Book in production to price securities, then the "1000" path sequence must also contain as a subset the smaller sequence (fewer paths) used to price securities. This is exactly what the comments in the above two calibration related files indicate. On the dates that this comment was added to these code files in 2002 and 2003, Defendants claim that they were using only LDS200, a sequence that mathematically cannot be a subset of any single seed 1000 path sequence and in particular, the Russell 1000 Path Sequence they claim to have been using despite the fact that the code could not generate it. Specifically, at the time they claim to have been using Russell 1000 Path Sequence, they claim that the production sequence used for pricing was only LDS200 which has 200 paths generated by multiple or mixed seeds. See Answers And Objections Of Defendants Citigroup Global Markets, Inc. F/k/a Salomon Smith Barney, Inc., And The Yield Book, Inc., F/k/a Salomon Analytics, Inc., To Plaintiff's Second Set Of Interrogatories, Requests For Production And Requests For Admission, Responses to Interrogatories 1 and 2, Ex. W. Mathematically, a 200 path sequence generated by multiple seeds cannot be a subset of any 1000 path single seed sequence. However, if one took the 32, 64 and 128 path stolen ACE sequences, added them together to get 224 paths and used the **REDACTED** that Defendants claim to have routinely used **REDACTED**, one gets **REDACTED** paths, a number approaching 1000 that is much more useful in calibration for triggering the options than any of the constituent smaller sequences, ACE 32, 64 and 128. The RCS code file paths.c,v (CGM 00176) around 2002 has a revision note indicating that **REDACTED** for "1000" paths²¹. This suggests that the mixed seed **REDACTED** **REDACTED** has been applied to the ACE combination sequence. For example an ACE combination sequence of 224 paths would be turned into a Super ACE sequence of **REDACTED** paths, assuming they started with the combination of three stolen ACE sequences. ACE 32, 64 and 128, and not some other combination. The referenced "1000" paths is close to the **REDACTED** paths of such a Super ACE sequence.

73. Further, the ACE 32, 64 and 128 path sequences are subsets of the Super ACE 896 path sequence. So if any of them are being used in production for pricing, the description fits perfectly. It is the only sequence identified in this case for which the description fits perfectly. It indicates that Defendants have used ACE in production for pricing and Super ACE for calibration for their own traders. See Section F below for more evidence of ACE use. The complete RCS code is necessary to determine how and when the ACE sequences were used by The Yield Book. Additional evidence that the Russell 1000 Path Sequence does not exist, is described below.

74. In the response dated January 25, 2006 to Plaintiff's Second Set of

²¹ Ex. AAAAA.

Interrogatories, the response to Interrogatory No.1: "Identify each sequence used in simulation by The Yield Book and made available to your clients, the time period for which each sequence was used and the principal author(s) of the sequence.", and No.2, "Identify each sequence used in simulation by your traders.", and No. 3, "Identify the code used to generate each sequences...", Defendants' do not mention that they used any sequence of 1000 paths, Ex. W. This is consistent with the fact that none existed at the time which was months prior to the development of the Radak 1000 path sequence in April 26, 2006. See September 7, 2007, Christopher Moore letter to Russell David Munves, Ex. Z.

75. Defendants asked Plaintiff to exempt the 1000 path sequence from production. Christopher Moore July 11, 2007 letter to Alan S. Loewinsohn, Ex. X. Plaintiff rejected their request. Alan S. Loewinsohn July 13, 2007 letter to Christopher Moore, Ex. Y. Nevertheless, Defendants' July 16, 2007 letter misrepresented to Magistrate Judge Pitman that Plaintiff had agreed to limit their production to sequences of path numbers of 200 paths or less (Ex. M), and produced no 1000 path sequences despite the July 3, 2007 Order to produce all sequences used after April 1999. On August 31, 2007, the Court granted Plaintiff's motion to compel production of the 1000 path sequence. In response, on September 7, 2007, Defendants produced the two 1000 path sequences (Christopher Moore letter to Russell David Munves, Ex. Z) generated by Radak Code rather than "unmodified" The Yield Book code. See Section B above.

76. Dr. Hayre, Defendants' head of Mortgage Research, testified and I concur, that any sequence must be tested for accuracy before being used for production. Hayre September 27, 2007 Dep. 72:7-20. Ex. AA. Yet Defendants claimed that the Russell 1000 Path Sequence was used in production for
 REDACTED
 REDACTED Russell testified that he
 REDACTED
 REDACTED

77. When Defendants produced the 1000 path sequences generated by the Radak Code, they misrepresented that the Radak Code was the "unmodified C source files, as extracted from the RCS archive ('monte.c' and 'gauss_random' files)". Defendants' September 7, 2007 letter to Plaintiff, Ex. Z. As discussed above, the code Dr. Radak used to generate the 1000 path sequences was all created by Dr. Radak, just like the other Radak Code he used to generate all the other Radak Sequences. It is clearly not "unmodified C source files, as extracted from the RCS archive".

78. At Dr. Radak's first deposition he could not say when or whether the Russell 1000 Path Sequence was ever implemented. Radak Sept. 28, 2007 Dep. 46:11-18, Ex. ZZZ. As noted above, at his August 29, 2012 deposition, Radak was allegedly tasked with REDACTED)
 after Russell's deposition on March 2, 2006. REDACTED
 REDACTED provide evidence
 of the fact that Defendants were using ACE, one would expect Defendants to be concerned that Plaintiff would discover the references to the non-existent 1000 path sequence and would want to try to provide a cover story for it in the event Plaintiff sought production of that sequence (which it ultimately did). Again,
 REDACTED

²² Russell Dep. 392:25-394:9. Ex. BB.

REDACTED

REDACTED Defendants installed the Radak 1000 Path Sequence on April 26, 2006. September 7, 2007, Christopher Moore letter to Russell David Munves, Ex. Z. Defendants story concerning the Russell 1000 path sequence simply is contradicted by all of the evidence. The story and the sequence were clearly fabricated for discovery.

79. A comparison of Post 2005 RCS files with the pre 2005 RCS files could provide additional evidence that the Russell 1000 Path Sequence never existed and is actually an intentionally misleading name followed by a functional description to remind them, without actually mentioning ACE, what was really going on, i.e., that they were using a Super ACE sequence, with the ACE sequences as subsets, rather than their alleged production sequence, the LDS200 sequence. Defendants could easily defeat the Super ACE contention -- except for the fact that Plaintiffs' contention is well-grounded -- by producing the post 2005 RCS files. Instead, they refused. The inference to be drawn from this alone is that they were using a Super ACE sequence for the following reason.

REDACTED

80.

REDACTEDRadak August 29, 2012 Dep. 34:14-22. Ex. CC. **REDACTED****REDACTED**

But he produced no development and testing files, claiming he deleted all the test files after he completed testing in April 2006 during this action so we cannot see **REDACTED**

REDACTED Accordingly, Plaintiff needed the updated RCS archive containing The Yield Book code used by Radak to do his 1000 path testing **REDACTED**

REDACTED In the alternative, Defendants, at minimum, could have provided a complete set of RCS files rather than the incomplete set produced to allow Plaintiff to compile an application to generate 1000 paths if The Yield Book could generate 1000 paths during the period when the Russell 1000 path sequence was alleged to have been used. The fact that Defendants refused to produce these documents, which were easy to produce and would have supported their claim of authenticity of the Russell 1000 Path Sequence, raises the inference that this evidence would have exposed the fact that the Russell 1000 Path Sequence could not be generated, that it was fabricated to cover up the use of a Super ACE sequence as shown above.

81. On August 29, 2012, Radak testified that he destroyed all of the sequence development files and testing files used in 2006 for the Radak 1000 path sequence, Radak August 29, 2012 Dep. 42:17-43:15, Ex. DD. Defendants directed Radak to develop a 1000 path sequence in March 2006, right after the Mr. Russell's first day of deposition on March 2, 2006. Russell at this 30(b)(6) deposition testified about the **REDACTED**

. So the destruction of the sequence development files and testing files is against Defendants' own policy. Russell Dep. 70:10-18. Ex. EE. Inexplicably Defendants did not follow the policy with respect to Radak 1000 Path Sequence which was clearly a major change - replacing one production sequence with another. Nor did Defendants instruct Radak to

²³ Plaintiff was forced to conduct the deposition in 2012 without the pertinent documents, e.g., the post-2005 RCS code file from which The Yield Book source code could be extracted that would allow Plaintiff to build The Yield Book application that would generate the Radak 1000 Path Sequence.

preserve the evidence for litigation, See Radak August 29, 2012 Dep. 37:7-20 and 42:17-43:15, DD and FF. Defendants confirmed that the Production contains no development file and no testing file for either of the two purported 1000 path sequences. This is exactly what one would expect if they were using Super ACE. One would not extensively test a non-existent sequence or one that was used to cover up a non-existent sequence.

82. The single-seed purported Russell 1000 path sequence created by Radak in 2007 (“Radak-Russell 1000 Path Sequence”) was, according to Defendants generated in accordance with one of two methods described by Russell at his deposition, both of which would result in the same set of numbers. See, August 18, 2011 letter of Christopher Moore to Russell D. Munves at fn 2, Ex. GG. One of the methods described involved generated a sequence of 200 paths and then generating a different sequence of 200 paths four more times. Since Russell was purportedly using the same seed as was used to generate the single seed 200 path production sequence, this 1000 path production was authentic, the first 200 paths generated would have to have the same numbers as the single seed 200 path production sequence. Yet my comparison of the single seed 200 path sequence produced revealed that they share no numbers in common with the Radak-Russell 1000 Path Sequence. See for example the first two pages of each sequence, the Production200 and Radak-Russell 1000, Ex. XXXX. Thus, it cannot be genuine. This further demonstrates that Defendants’ story about the Russell 1000 Path Sequence is completely made up to cover their use of Super ACE in calibration.

83. In the RCS files produced, The Yield Book code contains code commands to set the path numbers only for LDS100, LDS200 and Production200. It does not contain any command to set the path number to 1000. RCS files produced were last modified in May or July of 2005. The only other way the computer could ever receive an instruction to set the path number to 1000 is through customization by whoever starts up The Yield Book. That person would have to do so by either typing in the path number “1000” at the keyboard or by using a script file that calls The Yield Book and inserts the number 1000 using the same function at the launching of the application. This was confirmed by Radak in his August 29, 2012 deposition. In the code file main_new.F,v containing versions of this file that were used beginning in 1995, Ex. II, the relevant code under the option “-pnnn” custom-setting the path numbers have command lines, not just comments, that strictly limited any path number to “3 digits or less”. Radak Dep. August 29, 2012 110:2-111:6 and 113:23-116:8.. Ex. HH. It is also found in main_new.c,v containing versions of main_new.c used in 2005, Ex. JJ.

84. For example, in the Fortran code main_new.F, Ex. II, the code `if(buffer(1:2).eq. '-p')` checks that first two characters, the flag, is “-p”; then the next 3 characters are the 3 digits of the path number. The code `if(buffer(6:MAX_ARG_SIZE).ne.' ')` checks the 6th character, which is the 4th digit of the input path number, to be a blank ' '. If not blank, The Yield Book will print out error message 'INVALID OPTION: -p option must be followed by three or fewer digits' will print into computer screen if you try to input path number 1000. This code shows that The Yield Book could not generate 1000 paths during the period when the purported Russell 1000 path sequence was allegedly in use. It means that the Radak-Russell 1000 Path Sequence produced by Radak is as phony as a \$3 bill.

85. To put the code file main_new.F in perspective, it contains the main function for The Yield Book application mortopt, which is used for the pricing of all MBS, IO’s and PO’s, see

doc.2f,v from (CGM00176), Ex. KK. Doc.2f,v provides an overview of how The Yield Book code is structured and works. The limitation of 3 digits or less in `main_new.F` means that any sequences used for pricing of any MBS, IO's and PO's cannot have 1000 paths. When the calibration is performed, The Yield Book must also use the code `calibrate.c` (CGM00176). One key step of calibration to verify that the parameters of term structure model are correct, is to price certain IO's, and PO's, and match the prices generated by the model to the market prices for the same instruments:

```
fprintf(stderr, "Model_priceof %25s=%8.3f\n", bondname[deal]
[bond], mkt_price [deal] [bond] [j]);
```

`Calibrate.c,v` at lines 1148-1149 (CGM00176). Since at this stage of calibration and before Defendants used Super ACE, the sequence used for calibration was the same as the sequence used for pricing (as Dr. Radak confirmed he did when testing his 1000 path sequence, Ex. R), and in particular for pricing of IO's and PO's, Defendants had to use the `mortopt` application which uses `main_new.F` and so calibration must use a sequence whose number of paths is no more than 3 digits. So any sequence generated by The Yield Book during the period when the Russell 1000 Path Sequence was purportedly in production cannot have 1000 paths. But it could have 896 paths necessary for the Super ACE sequence described above, at least for the code running "nts" (new term structure model) code, for Defendants' internal traders. Predictably, the code reading in Super ACE was not produced by Defendants. As shown in Section F, Defendants have left out the code used to generate interest paths they used with ACE.

86. So while we know the Radak Sequences cannot be generated by The Yield Book (see Section B as well), the complete current RCS code should be produced to allow Plaintiff to build The Yield Book executable code to verify what sequences it can generate. Unfortunately, Defendants have successfully withheld this code. If they had nothing to hide, they would have produced it.

D. Defendants' Deficient and Altered Sequence File and Testing File Production

For 6 years Defendants had resisted the production of sequence development files and testing files, until Judge Swain directed them to produce them; they then made a production of 5 DVD's, CGM07077-7081 ("Development Production") on March 11, 2011. After I reviewed the Development Production and the transcripts of deposition of Teytel on August 26, 2012 and Radak on August 29, 2012, I understand Defendants' resistance to this production, it contains no verifiable evidence of independent development of sequence.

Defendants destroyed the code that might reveal the algorithm used in selecting the seeds, i.e., selecting sequences for LDS64 and LDS100. During discovery, they destroyed all the original development files and testing files for LDS200 and the Radak 1000 Path Sequence of April 2006. Defendants represented that all the files in the 4 out of 5 DVD's, CGM07077-7080, were for LDS200 and contained files under a single directory "lds200". Teytel changed all the "last modification date" of all files in the 4 DVD to July 2005, even he purportedly developed LDS200 from later 1999 to April 2000. His modification of the dates covered up the fact that he

fabricated the Development Production.

Defendants represented that they produced no development files and testing files for LDS64, LDS100 or either of the two 1000 path sequences. Disk CGM07081 contains a somewhat cryptically named directory called REDACTED which unlike the "lds200" directories on the other disks, does not indicate what sequence development it relates to if any. At his deposition, Dr. Teytel only very reluctantly admitted that this directory contains REDACTED REDACTED Ex. DDDDD. Only this disk contains files with last modification dates in 1999. These files are the lone survivor from Defendants' file destruction. Why would Defendants spare this code and produced it? The answer was provided by Dr. Teytel, who truthfully testified.

REDACTED As shown by a mountain of evidence in my first Expert Report and 2nd Declaration (Ex. G and E), the real algorithm for seed selection targeted ACE. Production of the seed selection files would be incriminating, but the REDACTED is clean, so it was produced in an effort to provide the appearance of a good faith effort with respect to the Development Production. But it does not succeed. This REDACTED production simply raises more questions.

It raises the question of why did Dr. Teytel also destroy all the later code files for REDACTED for LDS100, which also would not reveal the incriminating Algorithm, while producing the earlier REDACTED The answer is provided by Teytel's Notebook, which mentions ACE as many as 12 times in the names of code files relating to the testing of LDS100. This shows that Teytel was testing ACE against various REDACTED of LDS100.

If Defendants had made clear that they produced only the innocent REDACTED REDACTED but destroyed everything relating to LDS100 whose development followed LDS64, it would have been starkly obvious that Dr. Teytel had carefully selected out the incriminating documents for destruction. In light of this, it is not surprising that detailed analysis below shows that the purported development files for LDS200 in the Production were fabricated.

D1. Defendants destroyed all the original code relating to seed selection algorithm

87. The central issue that needs to be resolved to determine whether Plaintiff's ACE Derivative Use Claim or Defendants' denial and defense of independent development is correct is, obviously, the real algorithm that Defendants used to select their LDS sequences ("Real Algorithm"). Defendants admitted that their sequences selection is made by selecting the seeds. My first Expert Report and 2nd Declaration (Ex. G and E) concluded, based on a mountain of evidence, that Real Algorithm targets ACE. Defendants claim that it is the Mixed Seed Algorithm by Teytel (Ex. EEEE). As such, their code relating to seed selection algorithm ("Seed Selection Code") is a key piece of evidence to liability. However, Defendants fought tooth and nail for 6 years refusing to produce sequence development files and testing files including the Seed Selection Code, relying on the astonishing claim that they are "totally irrelevant". After the Court ordered production, on March 11, 2011 they made the Development Production.